# GNOME

Base GNOME packages for the full GNOME experience. Bundle with other packages to prevent package conflicts providing the same functionality.

> **TIP:** Include any and all packages you want installed in a list to `pacman`. That way `pacman` will resolve package dependencies correctly and not install packages that would cause conflicts with other packages later on in the setup; e.g. the `gnome` group installs `pulseaudio`, but `pulseaudio` and `pipewire` (see below) are conflicting packages, meaning they can't both be installed at the same time prompting you to remove one or the other. Explicitly selected packages take precedence over packages auto-selected via dependencies.

```
pacman -S gnome gnome-extra
```

# Setting up display manager

## Start GDM on boot

Start the GNOME Display Manager (GDM) on boot to be presented with a graphical login screen.

```
systemctl enable gdm
```

### When using NVIDIA proprietary drivers

For the longest time NVIDIA only supported their EGLStreams interface for Wayland sessions. Despite GNOME having support for both EGLStreams and the more popular GBM interface, the GNOME Display Manager disables the Wayland session via a `udev` rule, if it detects the proprietary driver is in use, to prevent problems with the login screen not showing.

To force enable GNOME's Wayland session even with the proprietary NVIDIA driver installed, check the following files:

- `/etc/gdm/custom.conf`: Make sure the line `WaylandEnable=false` is **commented out** (should be by default)
- `/usr/lib/udev/rules.d/61-gdm.rules`: Rename the file and create a symbolic link to `/dev/null`

```
ln -s /dev/null /usr/lib/udev/rules.d/61-gdm.rules
```

Keep in mind that Wayland depends on Kernel Mode Setting to function properly, so it is necessary to include the appropriate kernel modules in the initramfs and setting the kernel commandline parameter to enable KMS support for the proprietary NVIDIA driver!

See Graphics Cards on how to set up early KMS with the proprietary NVIDIA driver.

# Set Keymap for GDM

> **NOTE:** Executing this command while `chroot`ed into an installation will produce an error that the locale could not be found. Set after rebooting the system, press `CTRL + ALT + F3` when GDM shows up (or any F-key between 2 and 7) to switch tty, log in via the command line and execute the command as `root`.

```
localectl set-x11-keymap de
```

See instructions at Plymouth page on how to set up Plymouth.

# Misc additional packages

Additional packages you might want:

| Name | Description |
|---|---|
| `gthumb` | Image viewer with simple editing capabilities |
| `lollypop` | Music player for GNOME |
| `seahorse` | Secrets manager (login credentials, SSH keys, GPG keys) |
| `fwupd` | Firmware update manager; allows UEFI capsule updates in GNOME Software if supported by firmware |

```
pacman -S gthumb lollypop seahorse fwupd
```

# GNOME Keyring

Gnome Keyring is a useful tool for securely storing and managing passwords, SSH keys, and other sensitive information.

As `gnome-keyring` is already a member of the `gnome` package group, it should already be installed.

To manage the contents of `gnome-keyring` install `seahorse`:

```
pacman -S seahorse
```

# SSH Keys

You can use `gnome-keyring` to store the passphrases of your SSH keys for passwordless authentication. However, the systemd user service file does not include the `ssh` component.

To include the `ssh` component, edit the systemd unit file:

```
systemctl edit --user gnome-keyring-daemon
```

This opens an editor with a temporary drop-in file. In here you can override the parameters of the systemd unit file:

> **NOTE:** To properly replace a parameter, it first needs to be empty and then repeated on the next line with the value you want it to have. Additionally, setting the `SSH_AUTH_SOCK` environment variable will make applications aware of an already unlocked SSH key.

```
[Service]
ExecStart=
ExecStart=/usr/bin/gnome-keyring-daemon --foreground --components="pkcs11,secrets,ssh" --control-directory=%t/keyring
```

Save the file and close the editor. It will get automatically reloaded by systemd if necessary.

In order to reveal to applications that an SSH key has already been unlocked, set an environment variable for the current user:

```
echo SSH_AUTH_SOCK=$XDG_RUNTIME_DIR/keyring/ssh >> ~/.config/environment.d/envvars.conf
```

Re-login for the changes to take effect.

# Uniform application styles

## Qt applications

To make Qt/KDE applications fit in with the GNOME desktop you can install an Adwaita Qt theme and window decorations:

```
yay -S adwaita-qt{5,6}-git qadwaitadecorations-qt{5,6}
```

Then set the following environment variables in `~/.config/evironment.d/qt.conf`:

```
QT_WAYLAND_DECORATION=adwaita
QT_STYLE_OVERRIDE=Adwaita-Dark
```

## GTK3 applications

There is an Adwaita theme that brings GTK3 apps in line with the current LibAdwaita theme:

```
pacman -S adw-gtk-theme
```

Then open GNOME Tweaks and set the application theme for legacy applications to `adw-gtk3` or `adw-gtk3-dark` (for apps that are not dark mode aware).

## Firefox

Screenshot of Firefox GNOME Theme

Firefox can be customized to look like a GNOME native application by applying a GNOME Theme to it.

The simplest way to apply the theme is by installing *Add Water*, an application that allows you to install/remove the Firefox GNOME theme with a single click. It also allows to customize the GNOME theme with several options. It can auto-detect different versions of Firefox (repo package, Flatpak, Snap) as well as Firefox forks (Floorp, Cachy, LibreWolf).

```
flatpak install dev.qwery.AddWater
```

> **NOTE:** When Firefox receives an update the theme can break in some ways. When this happens you can uninstall the theme temporarily or hold off on updating Firefox until an updated version of the theme becomes available.

## Steam

Screenshot of Adwaita Steam Theme

There is an Adwaita theme available for Steam to make it fit in with the rest of the GNOME desktop. An app is available that can install and manage the theme for you:

```
flatpak install io.github.Foldex.AdwSteamGtk
```

# Remove potentially unwanted packages

## GNOME Dev Tools

```
pacman -Rsc gnome-{builder,devel-docs,multi-writer,terminal} accerciser d-spy devhelp glade sysprof
```

## User Software

```
pacman -Rsc gnome-{notes,recipes,sound-recorder} polari
```

## Games

```
pacman -Rsc gnome-{2048,chess,games,klotski,mahjongg,mines,nibbles,sudoku,taquin,tetravex} hitori iagno lightsoff quadrapassel tali
```

# Replace repo packages with Flatpaks

If you wish to use the Flatpak versions of packages that the GNOME desktop team maintains themselves, you can uninstall the packages that are available as Flatpak in GNOME Software.

## Remove

> **TIP:** Put substrings of package names between curly brackets `{ }` so the shell substitutes the values, e.g. `gnome-{calculator,calendar,characters,clocks}` is interpreted as if you typed `gnome-calculator gnome-calendar gnome-characters gnome-clocks`. Nesting also works!

Packages part of the `gnome` package group:

```
pacman -Rn gnome-{calculator,calendar,characters,clocks,connections,contacts,font-viewer,logs,maps,music,text-editor,weather} \
  baobab epiphany evince loupe simple-scan snapshot sushi totem
```

Packages part of the `gnome-extra` package group:

```
pacman -Rn accerciser cheese d-spy dconf-editor devhelp endeavour eog evolution geary ghex gitg glade \
  gnome-{2048,boxes,builder,chess,devel-docs,dictionary,games,klotski,mahjongg,mines,multi-writer,nibbles,notes,photos,recipes,sound-recorder,sudoku,taquin,terminal,tetravex,tweaks} \
  hitori iagno lightsoff polari quadrapassel seahorse sysprof tali
```

## Reinstall

Install the core GNOME apps as Flatpaks:

> **NOTE:** Some of the repo packages of the `gnome-extra` group have been discontinued by GNOME. The following Flatpak selections replaces these with actively developed alternatives.

```
flatpak install flathub org.gnome.{Calculator,Calendar,Calls,Snapshot,Characters,clocks,ColorViewer,Connections,Contacts,baobab,SimpleScan,Evince,Extensions,font-
viewer,Loupe,Logs,Maps,Music,NautilusPreviewer,TextEditor,Showtime,Weather,Epiphany}
```

Selection of Flatpaks previously from the `gnome-extra` package group:

```
flatpak install flathub
org.gnome.{Evolution,Geary,GHex,gitg,Glade,Boxes,Photos,seahorse.Application,World.Iotas} ca.desrt.dconf-
editor io.github.alainm23.planify page.kramo.Cartridges page.tesk.Refine com.mattjakeman.ExtensionManager
io.gitlab.adhami3310.Impression
```

For a list of additional packages, see GNOME Flatpaks.

Revision #27
Created 31 August 2021 12:32:03 by Sebin
Updated 16 February 2025 22:24:58 by Sebin